Beginners' Introduction to HTML

HTML (like most other SGML applications) helps you describe the *structure* of your document in a way that is portable from one machine to another. It does this by using *tags* to surround important *elements* of text so that the computer can recognize them.

How to write tags

Here's an example of a tag in use:

```
<title>My first attempt</title>
```

The tag name goes inside angle brackets before the text, and it is repeated after the text with a slash after the opening angle bracket. These are called *start−tags* and *end−tags* and the text they surround is called the *content*.

A few tags are defined as *empty*: they only have a start−tag and they don't enclose any content, for example

```
<isindex>
```

defines the current file as a searchable index: there isn't any text to surround, so there's no end−tag.

It is possible to omit some of the end−tags in some restricted circumstances, for example when a *list item* is followed directly by another *list item* you can type
first item
second item
but it is good practice to be orthogonal unless (a) you know how SGML works and when HTML allows you to omit end−tags or (b) you are using a conformant SGML editor which can handle this kind of minimization for you.

The file header

A HTML file should be self−documenting, so it should begin with a *header* which specifies that this is HTML, and gives the *title* of the document and links it with the *owner*. The header is followed by the *body* which is where your text goes:

```
<html><head><title>How to make $1,000,000</title>
<link rev="made" href="mailto:JillDoe@wunderkind.ulr.edu">
</head>
<body>
...
</body></html>
```

This structure should occur in every file so that you know what it is and who is responsible for keeping it up−to−date. There are a few other tags which can be included for special effects which we'll come on to later. A few important points to note:

1. the <html>...</html> tag surrounds everything in the file.
2. the <head>...</head> element surrounds both the <title>...</title> element and the <link> tag;
3. the <title>...</title> element surrounds some text which you make up (here, 'How to make $1,000,000');
4. the <link> element is empty (there's no </link>), but instead it includes some extra information *inside* the angle brackets which attributes the ownership. The rev="made" is required as shown, but you must substitute your own electronic mail address in the href="mailto:..." attribute;
5. the <body>...</body> element follows straight after the header and doesn't finish until just before the end of the file at the </html>.

Try typing the first few lines of a file in this form on your own machine, substituting your own document title and your own e–mail address.

The original HTML also defined <nextid>, <base> *and* <plaintext> *but these are deprecated and will be dropped in HTML+.*

The <isindex> tag mentioned earlier can go in the header if you want the document to be searchable. If you use it, put it somewhere like right after the closing </title> tag. The behaviour of browser clients varies from implementation to implementation when they encounter this tag. Mosaic inserts a prompt and a panel for the user to type the search string, some others (Lynx, for example) need you to press a key (S) first.

You can put comments in your file which you can see when editing it, but which won't get displayed to others. A comment looks like a tag but has no name: instead there's a !-- after the opening angle bracket and a -- before the closing one (and no end–tag). The comment text in between can go over many lines:

```
<!-- Here's a comment which I can see when I edit the file but
     which is invisible to anyone reading the file via a browser.-->
```

Paragraphs

Inside the body of the document, the most common element is probably the paragraph. In the original HTML, <p> is specified without a </p> end–tag, and is used to separate paragraphs, and most browsers still accept this usage.

It is more normal SGML practice to use <p>...</p> to *enclose* paragraphs, like this:

```
<body>
<p>Try typing a paragraph of your document. Put it between the
start-tag and end-tag for the body of the document, as this
one is.</p>
</body>
```

and HTML+ defines <p> in this manner. You can have as many paragraphs as you want, one after another, each one inside its own <p>...</p> tags.

You cannot use blank lines on their own to separate paragraphs, as you do in a wordprocessor: SGML pays no attention to multiple blanks, tabs and linebreaks (except in special circumstances)– to make your text format correctly you use the tags.

Section headings

Most documents come divided into some form of sections, each with its own heading. HTML allows you up to six levels of section heading, called <h1> to <h6>. Client programs (browsers) usually display different sizes, colors or positions of type for the headings. Here's a top–level heading:

```
<h1>Jill Doe - My Life</h1>
```

which would appear like this:

Jill Doe – My Life

Second–level headings (and all the rest) are done in a similar way:

```
<h2>Chapter 1 - Born to rule</h2>
```

which looks like this:

Chapter 1 – Born to rule

and so on.

Section headings in HTML are *section levels*, **not** *section numbers*, so <h3> means 'heading level

3', not 'section number 3': there is no automated section–numbering in HTML. You can have up to six levels of headings: but each level can *occur* as many times as necessary.

Accents

Most W3 clients (browsers) support the ISO Latin–1 *character entity* accents defined by the International Standards Organization (there are lots of others but they are not defined in HTML). They have to be typed in a special form, so that they work on all computers, because each computer manufacturer has his own (usually non–standard) idea about how to do accents, and none of them match! **Don't be tempted to use your computer's own idea of accented letters, because they may be complete garbage to other users on different computers.**

The form they take is a mnemonic for the name of the accent enclosed between the two characters `&` and `;` (ampersand and semicolon) like this: `é` so to get 'Resumé' you type:

```
Resum&eacute;
```

It's a bit longwinded if you don't have an SGML editor, but it's the only way to make sure your accents work on other people's machines: see the full list for more of them.

There are a few additional character entities for doing other stuff:

- `&` gets you an ampersand (&)
- `<` gets you the less–than (<) sign
- `>` gets you the greater–than (>) sign

There are no currency symbols except the regular dollar sign (just `$`) and no editorial marks like dagger or pilcrow.

Mosaic provides a Multi–Locality enhancement to make use of various national character sets in the X version (2.*) but this is not defined in HTML.

And now . . .

If you've grasped that lot, you should be able to write a simple sectioned document:

```
<html><head><title>How to make $1,000,000</title>
<link rev="made" href="JillDoe@wunderkind.ulr.edu"></head>
<body><h1>How to make a million dollars</h1>
<p>When I sat down to try and tell the world how they too
could become millionaires, everyone said 'John, you must be
crazy! How could you possibly give away a secret like that?'</p>
<p>Well, I decided that this kind of information was for
sharing; it really wouldn't be right to keep it to myself.
So here's my book, and I hope it makes you as much money as
it's making me.</p>
<h2>How to make money</h2>
<p>Write a book that tells everyone how to do it, and make
your money from the sales of the book. Like PT Barnum said,
'there's a sucker born every minute' - <em>stultus omni momento
nascitur</em> if you prefer!</p>
</body></html>
```

Remember, for normal text, formatting is irrelevant: browsers do the formatting for you, using the tags as guides. Give it a try, compare it with this one and come back for the next part when you're ready, where we'll look at lists, links and visual effects.